COMP ... & TECHNOLOGY

A GUIDE TO MAJOR JOB ACCOUNTING SYSTEMS:

The Logger System of the UNIVAC 1100 Series Operating System

NBS Special Publication 500-43
U.S. DEPARTMENT OF COMMERCE
National Bureau of Standards

NATIONAL BUREAU OF STANDARDS

The National Bureau of Standards[1] was established by an act of Congress March 3, 1901. The Bureau's overall goal is to strengthen and advance the Nation's science and technology and facilitate their effective application for public benefit. To this end, the Bureau conducts research and provides: (1) a basis for the Nation's physical measurement system, (2) scientific and technological services for industry and government, (3) a technical basis for equity in trade, and (4) technical services to promote public safety. The Bureau's technical work is performed by the National Measurement Laboratory, the National Engineering Laboratory, and the Institute for Computer Sciences and Technology.

THE NATIONAL MEASUREMENT LABORATORY provides the national system of physical and chemical and materials measurement; coordinates the system with measurement systems of other nations and furnishes essential services leading to accurate and uniform physical and chemical measurement throughout the Nation's scientific community, industry, and commerce; conducts materials research leading to improved methods of measurement, standards, and data on the properties of materials needed by industry, commerce, educational institutions, and Government; provides advisory and research services to other Government Agencies; develops, produces, and distributes Standard Reference Materials; and provides calibration services. The Laboratory consists of the following centers:

Absolute Physical Quantities[2] — Radiation Research — Thermodynamics and Molecular Science — Analytical Chemistry — Materials Science.

THE NATIONAL ENGINEERING LABORATORY provides technology and technical services to users in the public and private sectors to address national needs and to solve national problems in the public interest; conducts research in engineering and applied science in support of objectives in these efforts; builds and maintains competence in the necessary disciplines required to carry out this research and technical service; develops engineering data and measurement capabilities; provides engineering measurement traceability services; develops test methods and proposes engineering standards and code changes; develops and proposes new engineering practices; and develops and improves mechanisms to transfer results of its research to the utlimate user. The Laboratory consists of the following centers:

Applied Mathematics — Electronics and Electrical Engineering[2] — Mechanical Engineering and Process Technology[2] — Building Technology — Fire Research — Consumer Product Technology — Field Methods.

THE INSTITUTE FOR COMPUTER SCIENCES AND TECHNOLOGY conducts research and provides scientific and technical services to aid Federal Agencies in the selection, acquisition, application, and use of computer technology to improve effectiveness and economy in Government operations in accordance with Public Law 89-306 (40 U.S.C. 759), relevant Executive Orders, and other directives; carries out this mission by managing the Federal Information Processing Standards Program, developing Federal ADP standards guidelines, and managing Federal participation in ADP voluntary standardization activities; provides scientific and technological advisory services and assistance to Federal Agencies; and provides the technical foundation for computer-related policies of the Federal Government. The Institute consists of the following divisions:

Systems and Software — Computer Systems Engineering — Information Technology.

[1]Headquarters and Laboratories at Gaithersburg, Maryland, unless otherwise noted;
mailing address Washington,D.C. 20234.
[2]Some divisions within the center are located at Boulder, Colorado, 80303.

The National Bureau of Standards was reorganized, effective April 9, 1978.

COMPUTER SCIENCE & TECHNOLOGY:

A GUIDE TO MAJOR JOB ACCOUNTING SYSTEMS:

The Logger System of the UNIVAC 1100 Series Operating System

J. M. Mohr, A. K. Agrawala, and J. F. Flannagan

Department of Computer Science
University of Maryland
College Park, MD 20742

Sponsored by:

Institute for Computer Sciences and Technology
National Bureau of Standards
Washington, DC 20234

U.S. DEPARTMENT OF COMMERCE, Juanita M. Kreps, Secretary

Jordan J. Baruch, Assistant Secretary for Science and Technology

NATIONAL BUREAU OF STANDARDS, Ernest Ambler, Director

Issued December 1978

Reports on Computer Science and Technology

The National Bureau of Standards has a special responsibility within the Federal Government for computer science and technology activities. The programs of the NBS Institute for Computer Sciences and Technology are designed to provide ADP standards, guidelines, and technical advisory services to improve the effectiveness of computer utilization in the Federal sector, and to perform appropriate research and development efforts as foundation for such activities and programs. This publication series will report these NBS efforts to the Federal computer community as well as to interested specialists in the academic and private sectors. Those wishing to receive notices of publications in this series should complete and return the form at the end of this publication.

National Bureau of Standards Special Publication 500-43
Nat. Bur. Stand. (U.S.), Spec. Publ. 500-43, 79 pages (Dec. 1978)
CODEN: XNBSAV

Library of Congress Catalog Card Number 78-600135

U.S. GOVERNMENT PRINTING OFFICE
WASHINGTON: 1978

For sale by the Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402
Stock No. 003-003-02003-6 Price $2.50
(Add 25 percent additional for other than U.S. mailing).

A GUIDE TO MAJOR JOB ACCOUNTING SYSTEMS: THE LOGGER SYSTEM OF THE UNIVAC 1100 SERIES OPERATING SYSTEM

by

J. M. Mohr; A. K. Agrawala; and J. F. Flannagan
University of Maryland

ABSTRACT

This report has been prepared to serve as guides to the use of Logger, the job accounting system supplied by UNIVAC for its 1100 Series Operating System, Level 32. Logger provides a capability for the automatic collection of information that may be used both for billing a computer installation's customers on the basis of resources utilized by their programs, and for gaining useful insights into the performance characteristics of the system itself. This report describes the structure of the accounting log system, provides a description of the information contained in the log tapes, and describes how the information is gathered by the Operating System.

Keywords: Computer performance analysis; computer performance measurement; EXEC-8; job accounting systems; Logger system; resource utilization measurement; standard unit of processing; SUP; UNIVAC 1100 Series Operating System.

The research reported herein was supported in part by the National Bureau of Standards under Grant 5-9017, and in part by the Control and Automation Branch, Engineeirng Division, National Science Foundation, under Grant GK-41062 to the Laboratory for Pattern Analysis, Department of Computer Science, of the University of Maryland.

PREFACE

The usefulness of vendor-supplied job accounting systems in improving the performance of computer installations has long been recognized. Originally designed for user chargeback, these supervisor-resident programs produce data on workload and resource utilization that are essential to the estimation of installation efficiency. They are comprehensive, convenient to use and-- being bundled with standard-release operating system software-- essentially free of charge. Consequently, they have been used to forecast workload, drive prediction models, estimate capacity, reconfigure equipment, and set priorities for software optimization. They are arguably the most versatile and most widely used of performance measurement tools.

Taken as a group, job accounting systems have one additional characteristic that makes them particularly interesting to auditing and standards activities: they are nearly universal across contemporary medium-to-large scale computer installations. Some installations do not actively collect accounting data, some collect it but make no practical use of it, while others collect it exclusively to charge customers for their use of the computing resource. Yet nearly all have the capability of collecting this data and the potential for making productive use of it. It is not surprising, therefore, that federal managers have begun to turn to this tool as the basis for describing the performance of whole classes of computer installations, and even comparing performance from one site to another.

Several technical obstacles stand in the way of these expanded uses of accounting data, however. Job accounting systems reflect unique vendor ideas of hardware and software architecture, and different attitudes toward the accounting function. Consequently, they measure different things, measure the same things differently, and reflect a unique configuration of resource demands that may not be duplicated for the same workload on any other system. It is not always clear what specific system events are represented by accounting data, whether events are measured, inferred or prescribed, how precisely they may be measured, or what calibration may be needed to increase their accuracy. In short, the use of job accounting data beyond the individual installation level requires a detailed understanding of accounting systems as instrumentation device.

The study on which the following report is based was commissioned by the National Bureau of Standards in response to this need for better understanding of how accounting sysems work, what they measure and how well they measure it. It attempts to apply rigorous criteria of description, analysis and validation to an already valuable and potentially influential method of computer instrumentation.

This report covers the job accounting system for more than seventy large-scale UNIVAC computer systems installed in the federal government: the Logger System of the Sperry UNIVAC 1100 Series Operating System.

To managers and technicians at these and similar installations in the private sector, this report is recommended as preparation for using job accounting data for performance improvement, reporting, prediction or control. To other local installation personnel, it is proposed as a model for describing and validating their own job accounting system. To higher levels of ADP management and policy-making, it may provide some insight into the hidden pitfalls of depending on accounting systems for more general applicability than they were designed to support. Finally, it is intended neither as an endorsement of specific job accounting systems and the equipment that supports them, nor as a criticism of those systems for falling short of objectives they were never intended to meet. On the contrary, it is offered in the belief that accounting data gives us a useful but sometimes inaccurate window on computer performance, and is therefore worthy of a serious professional effort to interpret what it reveals to us.

Table of Contents

I. Introduction

Virtually all large third generation computer systems maintain an extensive record of the various user- and system-oriented events that take place during normal operations of the systems. This record is created by the system automatically (without any interaction from the user or the operator) in the form of an accounting log which consists of a sequence of entries, usually one per event. As this may require a rather large number of entries, the accounting log is often written out on magnetic tapes. In most installations, the log is used for accounting and billing purposes and therefore is often referred to as an accounting log. To allow for accurate user billing, the log has to maintain information about the computing resources used by each user program. The log may also maintain information on the reliability of the system and on the type of errors encountered by the system.

While the log may contain entries of interest to computer system operations and management personnel, the information on the log about the user resource loadings makes the log a valuable and inexpensive source of information for workload characterization studies [1], system analysis, and the design of charging algorithms.

Before using a source of information such as the EXEC-8* System log in any study, one needs to know exactly what information is recorded in the log entries and how this information is obtained. As the information for the entries may depend on various measured quantities, an analysis of the accuracy and validity of the values entered in the log would be necessary. Such an analysis of log entries is the subject of a subsquent report. The main purpose of this report is to examine the log produced by the UNIVAC 1100 Series Operating System in order to describe the types of information contained on the log tapes and how this information is gathered by the EXEC-8 system. Section II describes how the log entries are organized on the log tape and how the log tape itself is generated. The types of information available from the log entries and some information

----

* EXEC-8 is a registered trademark of the United Software Corporation and is the Operating System used on the UNIVAC 1100 series machines (UNIVAC 1106, 1108, and 1110). For the purpose of this report we will use EXEC-8 and "the Executive" interchangeably.

on the log entries themselves are presented in Section III. Sections IV and V deal with the quantities that are measured and computed by the system for accounting purposes, the meanings of those quantities, and the exact method by which they are measured or computed.

While this report is based on the UNIVAC system installed at Maryland University, we believe that most of the description is generally applicable to the current version of the 1100 Series Operating System (Level 32). Any information that is installation dependent or that can be altered at system generation time is noted as such in the text.

II. Organization of the Accounting Log

The system accounting log is organized as one sequential file that is composed of a series of entries each of which contains twenty-eight 36-bit words. There are up to 28 different types of log entries, or records, each corresponding to a different event that may occur in the course of processing a run. Typical events that may cause log entries to be written include the initiation or termination of a run or of program, the assigning or freeing of a mass storage file or of a tape drive, or recovery from a system crash.

A. Organization of the Log Tape

For each event that occurs , an entry is made in the log file that identifies the event, the time it occurred, the name of the run or task (program) in which it occurred, and certain other information that is dependent on the nature of the event.

The EXEC-8 Operating System at the University of Maryland allows for the concurrent execution of batch runs, demand runs, and real time tasks. As each run is processed, the log entries that are needed to record its appropriate events are generated. Information on the log tape is grouped by run and sequenced chronologically by run termination time. Within the information for each run, events are recorded in chronological sequence. Thus the accounting log provides a chronological record of certain events that contributed to the processing of a given run. For this same reason, however, it is difficult to reconstruct a system-wide chronology of events from the log file. If any two consecutive system events are charged to different runs, as is often the case in a multiprogramming job mix, the log entries for these events will not appear in sequence on the log tape but rather far apart and out of system chronological sequence. The

user interested in a system-wide chronology of events may be better advised to consider using the vendor-supplied System Instrumentation Package (SIP) for this purpose.

B. Generation of the Log Tape

As log entries are generated for a run, they are time stamped by the EXEC and placed at the end of a common linked list in core. When the number of entries being held in core reaches a limit which is preset at system generation time, the log records are placed into a linked list in one common catalogued file maintained by the Executive. This catalogued file allows the Executive to recover information generated prior to a system failure. Should a system failure occur, the recovery routine of the Executive would close all of the open log chains and would cause a type 18 log entry to be written onto the log tape.

When a run terminates, the Run Termination and Run Termination Supplement log entries are generated. The linked list is then searched for entries pertaining to the run that has just been completed. Those entries that are found are then written out to the accounting log tape.

III. Information Available from the Log Tapes

This Section provides a description of the classes of log entries and the categories of information they contain. Appendix A describes the format of each specific log entry and the specific information each contains.

A. Types of Entries

Although the entries are numbered from 1 to 28, not all types of entries are described in the Programmer Reference Manual [2]. The log entries not described in the Manual include types 20, 23, 24, 27, and 28. Types 20 and 28 are presently unused, and the others are described below in the entry class to which they belong.

There are five basic classes of entries recorded on the log tapes: resource utilization entries; message entries; run configuration change entries; error-record entries; and system action entries. Table 1 lists all entries in their numerical order and provides the class to which each is assigned.

Resource utilization entries accumulate for each run the amount of resources it used. These resources include:

* CPU time

* Executive Request (ER) and Control Card (CC) time

* I/O counts

* Voluntary Delay Time (VDT)

* Standard Units of Processing (SUP's)

* Core block SUP's

* Number of tracks used

Each of these items is described in more detail below.

Message entries are used to record communications between the user, the operator, and the system during processing. The various types of information messages are recorded in separate log entries.

Run configuration change entries are used to record information necessary to indicate that the configuration of a run was changed during processing. A run changes its configuration any time a disk file is assigned or freed, or anytime a tape drive is assigned or freed.

Error-record entries provide information on a wide variety of system errors such as system failures, I/O errors, errors encountered while loading EXEC segments, software errors, and hardware faults. These entries can be quite useful in determining the reliability of the system.

The type 24 log entries describe five software-detected error conditions that would have been handled as EXEC errors (system stops) in previous versions of EXEC-8. These errors are:

* mass storage block allocation bit table checksum error

* file conflict registering removable disk

* removable disk directory error

* removable disk cannot be registered

* illegal function or communications error.

The type 27 entry consists of 4 subtypes which are used on the Univac 1110 to report hardware faults. These faults are:

* IOAW storage parity check interrupt

* IOAU ACR parity check interrupt

* IOAU channel interface parity check interrupt

* CAU/IOAU interface parity check interrupt.

System action entries comprise the final class of entries, as noted above. These entries describe action taken by the system or user. An entry in this class might pertain to the start or end of symbiont processing. (The symbiont processes are a complex of Executive routines providing the user interface with unit record peripherals, for example, onsite and remote card readers and printers.)

Type 23 entries, which are used by the UNIVAC Software Instrumentation Package (SIP), belong in this category. Type 23 entries vary in length depending upon how many levels of SIP are enabled. The type 23 log entry is used only if SIP is set up to write its data to the accounting log file and is the only entry that is not limited to 28 words in length. SIP may also be configured to use its own files, in which case no records are written to the accounting log file.

B. Types of Information

There are four distinct types of information stored on the system log: recorded information; measured information; computed information; and information contained in the messages recorded during system operation.

1. Recorded information is generally static information that is merely recorded on the log tapes. Information of this type includes the identity of the runs, programs, their account numbers, etc. This information is supplied to the accounting log system by the user or by the system.

2. Measured information is information that is directly measured by the system and recorded on the log tapes. Items of this type include CPU time, the number of core blocks used, and similar measures of resources used.

3. Computed information is information calculated from measured values stored in the accounting log. The memory time integral is a good example of this type of information. (See page 10.)

4. Messages record the content of communications between the system and the operator or the user which may take place during processing. These messages are simply recorded as they occur in the course of a run.

Table 2 provides details on the information items recorded in the various log entry types. It will be noted that many items of infomation are recorded in more than one type of entry. Much of the information recorded on the log tapes is obtained from the Program Control Table (PCT) which is described in the next Section. This table contains much but not all of the information that EXEC-8 maintains about a run. Information such as the amount of unit record I/O performed for the user is not included in the PCT. Other information is written onto the log entries directly from control cards submitted for each run.

A few observations are in order concerning some of the measured and computed information items. First, the I/O counts are reported as being performed to devices classified into up to 10 groups. For accounting purposes all of the I/O devices on the system are divided into not more than ten groups with each group referring to a specific class of I/O device. The assignment of devices to groups is installation dependent. The group assignments for the I/O devices on the University of Maryland 1108 system are given in Table 3. It will be noted that only seven of the ten groups are used. In general, the devices are put into groups depending upon their speed, the fastest devices being in the lowest numbered group.

A second observation to be noted is that most of the recorded and measured information written onto the log tapes is cumulative in nature. For example, to determine the amount of CPU time used by a program, it is necessary to subtract the amount of CPU time reported in the program initiation log entry (type 16) from that reported in the program termination log entry (type 4). This is true for most of the measured and computed quantities.

The messages recorded on the log tapes may come from several sources. Either the user (via Executive Request @ LOG or @@ LOG) or the operator may enter messages into the log file. Also, messages sent by the system to the operator are recorded, as are operator replies to system messages. In addition, the system may write other messages onto the log tapes pertaining to such events as tape labelling or checkpointing.

C. Program Control Table

Before discussing the quantities that are recorded on the log tapes it is necessary to digress and describe how the system keeps track of all of the information that it collects about a run. Associated with each program is a table known as the Program Control Table, which ranges in size from 512 words to a site-configurable maximum of up to 21,000 words. The PCT contains most of the information that Exec 8 needs to maintain about a program or run in order to execute it. A PCT is defined for each program as a write protected static data bank. (At some installations, the user program may address the PCT directly by first executing an LDJ instruction using PCTBD as the bank name. The address of the first word of the main block of the PCT is defined as RPCTA$. Its address is normally 0776777 (octal). The Executive Request PCT$ may also be used to retrieve all or part of the PCT.)

As a program is being executed, information is accumulated in various entries of the PCT and written out to the log when the program finally terminates. Each time the program makes use of any system resources, the appropriate entries in the PCT are updated. The PCT also contains static information about the program and the run. This information includes the RUNID, ACCOUNT NUMBER, PROGRAM NAME, as well as information on all of the files that have been assigned. Most of the static information is obtained directly from the Program Control Table.

At some installations the Executive Request INFO$ may be used to obtain information about the amount of resources used by a run. This information includes CPU time used, I/O time used, and the memory time integral of the run.

IV. Measured Information

The EXEC-8 system keeps track of five measured quantities for each run. This information is accumulated during the execution of each program and kept in the PCT for the program. The measured quantities are either entered directly into the log

entries or are used in computing the values of calculated quantities which are then entered in the log entries. In this section, we examine the measurement technique used to obtain each of the measured quantities.

A. CPU Time

The CPU time used by a program is that amount of time that the program had control of the central processing unit as measured by the real time clock. The real time clock has a resolution of 200 microseconds and is used in the following way. At the time a program is given control of the CPU, a value is put into the Real Time Clock Register that is equal to the smallest of: the time quantum the program is to receive, the interval before the next event is to be removed from the timed wait queue, or the maximum value of the clock. Events on the timed wait queue may include periodic activities, activities waiting for some impasse to be removed, and entries for an I/O channel specifying a maximum time for transfer completion. During program execution, the contents of the lower half (bits 17-0) of the Real Time Clock Register are decremented by one every 200 microseconds, independent of program control or supervision. (Control unit time utilized for each decrementation cycle is 300 nanoseconds.) A real time interrupt is triggered when the control unit detects that the contents of bits 17 through 0 of the Real Time Clock Register have decremented from 000000 to 777776 (base 8), or from 0 to -1 (base 10).

During the period that it has control of the processor a program or activity may perform whatever processing it requires. This processing is suspended when a real time interrupt or any other interrupt occurs. whenever an interrupt occurs the program in control of the CPU is forced to relinquish its control of the CPU, and control is transferred to a routine for handling the specific condition that caused the interrupt. The system saves the value in the real time clock before beginning the actual execution of the interrupt handling routine. If the interrupt was caused by a condition requiring immediate CPU attention, the condition is handled. When the interrupt processing is completed, control is passed to the Dispatcher. The CPU time of the activity that had control of the CPU before the interrupt occurred is updated based on the time at which the interrupt occurred. (See Figure 1.) If no higher priority tasks have arrived since the last activity or program was scheduled and if that program has not completed its CPU quantum, that program will resume its control of the CPU with a quantum equal to the time remaining from its previous quantum.

The program that has control of the CPU at the time an interrupt occurs is not charged for the time spent by the CPU in servicing the interrupt. However, there is a certain amount of overhead that is involved in determining the nature of an interrupt, disabling other interrupts, recording certain information, and transferring control to the appropriate interrupt service routine. This process is known as "posting" the interrupt. The posting or recording process requires approximately 10 to 20 machine instructions. EXEC 8 (Level 32) does not keep track of this type of activity separately; thus the program in control of the CPU at the time the interrupt occurred is charged for the time involved.

The UNIVAC 1108 definition of CPU time depends only on the amount of real time a program had control of the CPU and not on the amount of processing it has done, or the number of memory cycles it has received. A somewhat different approach is used in the measurement of CPU time on the UNIVAC 1110. First, a measurement is made of Command Arithmetic Unit (CAU) time usage by a run. This measurement corresponds to the measurement of CPU time on the UNIVAC 1108 described above. This quantity is recorded in the type 17 log entry but is not used for billing purposes. The reason that this quantity is not an adequate measure of CPU usage is that the UNIVAC 1110 has two types of main storage, a 280-nanosecond read/480-nanosecond write primary memory and a 1.5-microsecond or 750-nanosecond read/write extended memory. If a program runs from primary memory, it should require approximately one-half to one-quarter as much real time as it would require to run from the extended memory. In order to make billing repeatable, the 1110 supplies two Storage Reference Counters (SRC) for each of the Command/Arithmetic Units in the configuration. One SRC counts references to primary storage while the second counts references to extended memory. When an interrupt occurs the counting of references by these two counters stops, and the contents of the primary storage reference counter and the extended storage reference counter are transferred to General Register Stack (GRS) at locations 056 (octal) and 057 (octal), respectively. The counting of storage references may be enabled by two privileged instructions. These reference counts are written onto the log tapes as the CPU time rather than as the CAU time. From these counts a more accurate measure of the CAU usage by a program can be obtained, at least in principle.

B. Core Blocks

A Core Block (CB) is a block of 512 contiguous words of main storage and is the smallest amount of memory allocated by the EXEC-8 system. The size of a program, and thus the number of core blocks that it requires, may vary during execution. Because of this, it is impossible to record the actual number of core blocks used, but rather a quantity that corresponds to a memory-time integral is recorded. This quantity is discussed in Sections V.B and V.D. Section V.B gives the definition of the unit of time that is used in this calculation. The instantaneous size of a program is given in word 231 (octal) of the Program Control Table.

Whenever the Dynamic Allocator (DA) [3] loads a program into memory, the number of blocks allocated and loaded is recorded in the PCT entry. Note that the Dynamic Allocator has the responsibility for making entries for the size of a program in word 231 (octal) of the PCT. Whenever a program changes size, the Dynamic Allocator is also called, as the program may have to be moved or swapped.

From the log tape data, it is possible to determine only the time average number of core blocks used by a program. No information is available as to the range of sizes or the distribution of sizes of the programs.

C. I/O Counts

The I/O transfer counts recorded in the Program and Run Termination entries in the log file do not record the actual number of words read or written by the Executive on behalf of user programs. The recorded value is computed based on the actual count of word transfers, modified by certain characteristics of the device the user program has requested (as described in Section V.A. below). The actual count of word transfers is kept by the accounting system until the relevant log entries are written. The count is accurate, within the following limitations. On a read operation, the count for the number of words actually transferred may be less than the number of words requested if an end of file is encountered before the requested word count has been reached. On a write operation, the number of words actually transferred should equal the number of words requested to be transferred. In the event that an error occurs during an I/O operation, the operation is automatically retried. The number of retries that may be made before the operation is aborted is determined by the type of I/O device involved. The

log system keeps count only of the number of words transferred on the last retry, whether it was successful or not, and not the actual number of words transferred during successive retries to satisfy the same request.

D. Tracks

Whenever a mass storage file is created or changes in size, this fact is recorded in the PCT of the program which performed this operation. A separate entry in the PCT is maintained for each file assigned to the run. The size of a file is kept in terms of granules. A granule is equal in size to a track or a position and is the basic unit in which mass storage is allocated.

E. Voluntary Delay Time

The Voluntary Delay Time (VDT) of a user program is defined to be the amount of time for which a program is blocked because of user-induced delays. For a batch job, the only source of voluntary delay time is a timed wait activity. Demand programs can accumulate voluntary delay time in two ways. They may request a timed wait just like a batch job or they may accumulate voluntary delay time during the time that the system is waiting for input from the user. This time is measured by a special timer in 200-microsecond units. The period starts when the run goes into an I/O wait for terminal input and ends after a line of input has been received and the program has been swapped in. Time spent waiting for a system response or in "waiting on facility" is not included in voluntary delay time.

The voluntary delay time is accumulated throughout the run, and appears in five different log entries: types 2,4,5,16, and 17. In the type 2 entry, the voluntary delay time is added to the total processing time of the run. The type 2 entry is the Facility Usage Log entry and can be used to determine the "chargeable" connect time of a device or a file. The user is charged for the amount of time required to run the program plus the amount of time his program was holding the resource but not doing any processing. Voluntary delay time is also used in the calculation of track seconds which are recorded in the type 5 log entry. (See section V.C.)

V. Computed Information

In addition to measured information, the log tapes record a variety of quantities that are calculated on the basis of measured quantities. In this Section, we examine the computed quantities, their definition, and the method of computation.

A. I/O Counts

As noted in Section IV.C. above, I/O transfer counts do not represent the actual number of words transferred by a user program. This recorded value is related to and based on the actual count of transfers, but is biased to reflect the overhead characteristics of the particular device type requested by the program. If the device type that a program requests is not available, for example, and the system assigns the file to an alternate device, the actual count of transfers is changed to the number that would have been required for the device type the program originally requested. The count is also adjusted to reflect the requested device-specific overhead functions, such as latency time for rotating mass storage devices, seek time for moving head devices, and startup time for tape drives. An example will serve to illustrate this point.

The University of Maryland configuration includes a number of rotating mass storage devices. For such devices, latency time is defined as the amount of time that it takes the requested data sector to reach the read/write heads after initiation of an I/O request. This quantity is assumed to be uniformly distributed on the interval (0,T) where T is the time required for one revolution of the device. The mean value of this distribution (T/2) is used as the latency time for each access. The seek time is the amount of time required to move a read/write head to the correct track. The distribution of this value depends upon the distribution of requests. For the purposes of the log tapes and the billing algorithm, a constant value S is used for this factor. Table 4 presents the latency time, seek time, and transfer rate for each of the rotating devices at the University of Maryland. The word count reported on the log tapes is the sum of the number of words transferred, the latency time T/2, and seek time S. Before this summary is done, the latency time and seek time are converted to words transferred. This is done on the following basis.

For each device we define R to mean the number of words transferred per microsecond. The value V added to word count is $V = (T/2 + S) * R$. This quantity represents the number of words

that could have been transferred in the time equal to the sum of the seek time and the latency time. Thus, the number of words the user is charged for transferring will change if the number of I/O accesses used to transfer a given number of words changes. For example, if 10,000 words were to be transferred in one access, the word count recorded on the log tape and the word count for which the user would be billed would be:

$$W = (T/2 + S)R + 10{,}000.$$

Whereas, if the same 10,000 words were to be transferred to the same device in one hundred 100-word transfers the word count would be:

$$W = 100(T/2 + S)R + 10{,}000.$$

Thus, it can be seen that the billable charges for the I/O counts can vary depending upon the number of transfers used.

B. Standard Unit of Processing (SUP)

The SUP is used as the unit of time in all of the time integrals calculated by the system. Univac defines a SUP as:

> A unit devised to provide a consistent measure of processing service as viewed by the user program. Input to the calculation of SUP's is weighted such that SUP's will, as nearly as possible, determine elapsed time to perform a unit of work in a serial environment on a unit processor with no overlap of I/O and CPU operations. [3]

The number of SUP's used by a program is calculated by using the following formula:

$$SUP's = (SRC \times X) + I/O(T) \times Y(D) + Z(D) + ER$$

where

SRC = Total storage references (1110) or CPU time (1106/1108)

X = SRC conversion factor (1110) or 1 (1106/1108)

I/O(T) = I/O words transferred for an I/O request

Y(D) = Conversion factor for I/O(T) based on

transfer rate for the requested device

Z(D) = Average access time for the requested device (latency and seek or startup)

ER = The sum of the fixed charges for ER service.

The value used in the SUP calculation for the Storage Reference Counts is the sum of the SRC's to primary storage and the SRC's to extended storage. The conversion factor, X, is the average speed of the memory. This insures that the charges for CPU will be the same regardless of where in main memory the program was loaded. Use of the SUP is an attempt to determine true monoprogramming costs.

SUP's are expressed as units of time; each SUP corresponds to 200 microseconds of processing time. Since CPU time as well as Executive Request and Control Card charges are already expressed in 200-microsecond intervals, these values are simply added to the SUP calculation.

The conversion factor for the Storage Reference Counts for the 1110 is such that processing time may be expressed in units of 200-microsecond intervals. The summation is made for I/O counts which are maintained separately for each device group; a value is added to the SUP count for each I/O operation performed.

The average latency time and seek time for a device are added together, and converted to units of 200-microseconds for use as Z(D). Y(D), the conversion factor for I/O(T), is the amount of time expressed in 200-microsecond units required to transfer one word to the device.

C. Track Seconds

For billing purposes online mass storage is charged at a fixed number of cents per track per day. This would not allow a user to be charged for temporary files since these are deleted at the end of the run. When a temporary file is assigned, the system records the number of tracks used and the length of time they were in use. Each time the size of a file changes this number is updated. Track seconds can be thought of as the time integral of the number of tracks used by a temporary file.

D. The Core Block SUP (CBSUP)

The CBSUP is calculated as the number of core blocks used by the program, times the number of SUP s used while the program was that size, summed over the various sizes of the program. This is to allow for the fact that the size of a program may vary during its execution. The use of CBSUP's is an attempt to measure use of all of the system's resources, such as core, processor time, and I/O. The CBSUP is the memory time integral mentioned in Section IV.B.

E. Executive Request (ER) and Control Card (CC) Charges

The final quantity that is computed is the value used for ER and CC charges. The control card charge is a fixed amount that is charged to process each control card that is submitted by a run. The amount the user is charged for processing varies and depends upon the control card entered. The charges used at the University of Maryland for the various control cards are given in Table 5. At Maryland, there are basically two classes of control cards, those that are "inexpensive" and those that are not. When using the inexpensive control cards, the charge in Table 5 is merely the charge for processing the control card by the system. The work required to perform the operation specified by the control card is done in user state and is charged to the user as user work. For the other control cards (@ASG, for example), the work required to perform the indicated operation is performed in "system state" and thus is billed to the user directly by the control card charge.

An Executive Request is a request made by a user for the Executive to perform some service or operation. The charges actually incurred during the processing of an ER are incurred by the system, since the system is in control of the CPU. Therefore, the system must pass these values on to the user, otherwise the user would not be charged for processing done on his behalf. Some of these operations, such as reading the time-of-day clock, take a fixed amount of time. The CPU charges for these requests are stored in a table and when the ER is made the charge is added to the user's totals. Other Executive Requests take a variable amount of time. For these requests, the user is billed for the amount of resources actually used.

It should be pointed out that no user is charged for the CPU time required to process an interrupt after the interrupt routine is entered by the system. The user is, however, charged for the I/O interrupts that his run generates. This is done by adding to

the ER charge for the user that initiated the I/O the charge for handling the interrupt at the end of an I/O operation. The units for the ER and CC charges are the same as for the CPU time, that is, intervals of 200 microseconds. Table 6 contains the charges used at the University of Maryland for all the fixed Executive Requests. It should be noted that both Table 5 and Table 6 specify the charges in instruction cycles. The EXEC converts these values to 200-microsecond units.

VI. Concluding Remarks

Although users of any large scale third generation computer system are directly affected by the accounting system provided by the system, few users study or fully understand it.

Before the information contained on the accounting log tapes can be used in any study, the sources and accuracy of the information must be studied. In this report we have attempted to describe exactly what information is available on the log tapes and the means by which this information is collected. It is hoped that this report will serve as a readable description of the accounting log system of EXEC-8, and that it will be of use to those who are planning studies in which the system log tapes will be used as the source of their required data.

While these reports confine themselves to the accounting log system of Univac's EXEC-8, we hope that they will inspire other groups to undertake similar studies on accounting log systems provided by other operating systems in general use. We believe that a collection of such studies would be extremely useful to those researchers who are examining the characteristics of user behavior on a wide variety of machines, as well as to those researchers who are performing studies on individual computer systems.

The authors gratefully acknowledge the assistance supplied by the University of Maryland Computer Science Center in obtaining the data used for this report. They would also like to express their appreciation to the staff at the Center and especially to Ira Gold for the help they received in preparing this work.

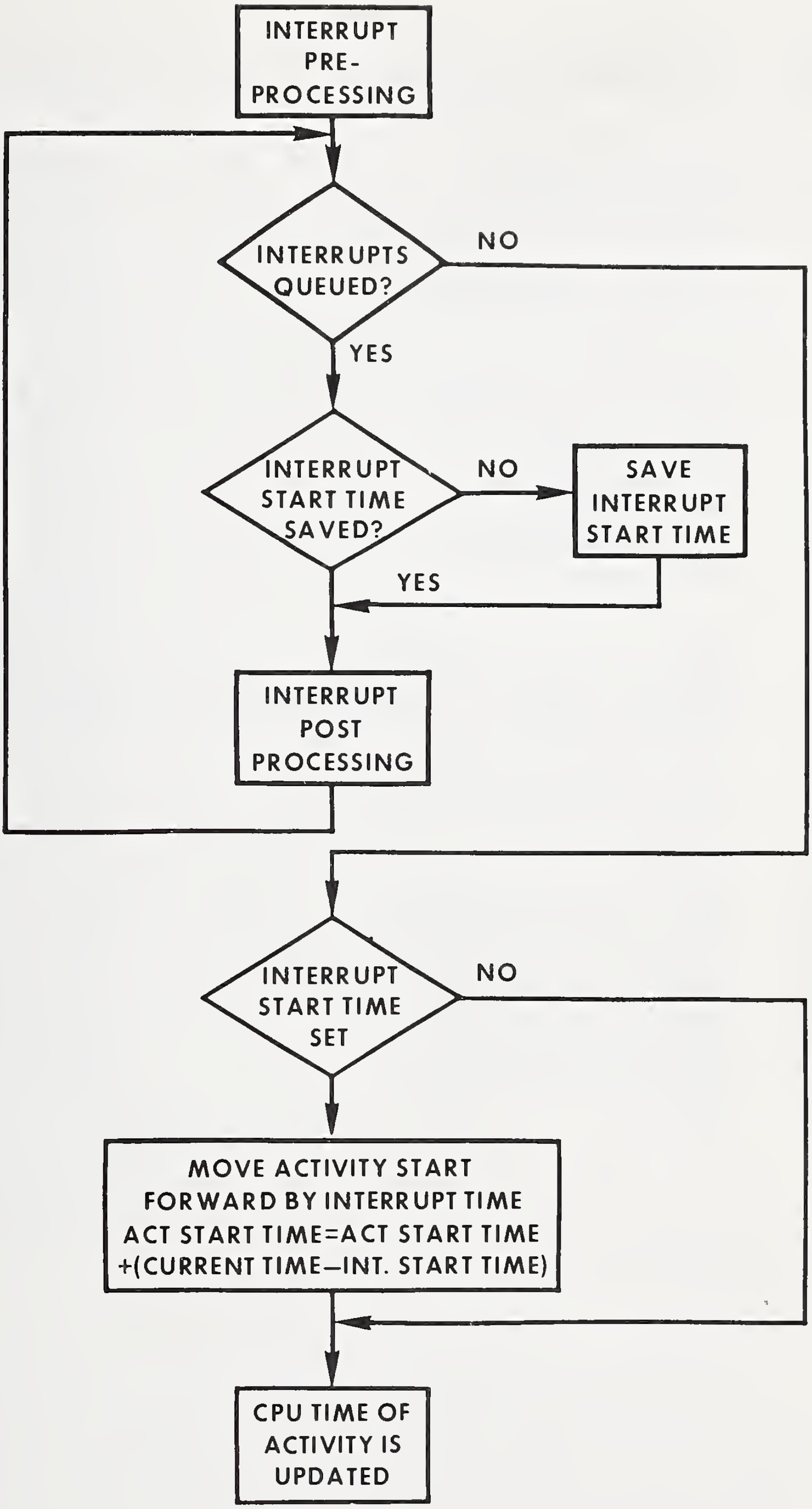

Figure 1. Calculation of CPU Time After an Interrupt.

Type	Description	Class
1	Control Statement	Message
2	Facility Usage	Configuration
3	Catalogued Mass Storage File Usage	Configuration
		Resource Utilization
4	Program Termination	Resource Utilization
5	Run Termination	Resource Utilization
6	I/O Error	Error
7	Console Log Entries	Message
8	Checkpoint Log Entry	Message
9	Run Initiation	Configuration
10	Console Replies	Message
11	Log Keyin	Message
12	Unsolicited Keyin	Message
13	Tape Labeling	Message
		Error
		Configuration
14	Symbiont End of Processing	System Action
15	Symbiont Start of Processing	System Action
16	Program Initiation	Resource Utilization
17	Run Termination Supplement	Resource Utilization
18	Recovery Close-out	Error
19	Cooperative Accounting	Resource Utilization
20	Unused	
21	Symbiont Close-out	System Action
22	EXEC Segment Validation	Error
23	Software Instrumentation Package	System Action
24	Software I/O Error	Error
25	Checkpoint SUP Charge	Resource Utilization
26	Restart SUP Charge	Resource Utilization
27	IOAU-CAU Hardware Error	Error
28	Unused	

Table 1. Log Entries by Type Number and Class

Information Item	Log Entry Number
Device Information	
Subsystem Number	2,6,24
Unit Number	2,6,24
Equipment Code	2,3,6
Reel Number, Pack-ID	6
Facilities Usage	
CPU Time (SRC's)	4,16,17,25,26
ER and CC Charges	4,16,17,25,26
SUP's	4,5,16,25,26
CBSUP's	4,5,16,25,26
Voluntary Delay Time	4,16,17,25,26
Unit Record I/O	5,14
Tracks * SUP's	5
SUP's + Voluntary Delay Time	2,3
I/O Transfer Counts	4,16,17,25,26
CAU Time (1110)	17
Symbiont Name	14,15
File Information	
Date and Time of ASG or FREE	2,3
Filename	3,24
Qualifier	3,24
Current Assigns	3
F-Cycle	3,24
Granule Counts	3,24
Mass Storage Address	22
Checksum Received/Expected	22
Sectors Allocated or Released	5
Allocation/Release Calls	5
Directory Control Calls	5
Track Seconds	5

Table 2. Log Entries by Information Items

Information Item	Log Entry Number
Messages	1,7,8,10,11,12,13,18
Program Information	
Program Name	4,16
Version Name	4,16
Program Initiation Date/Time	16
Program Termination Date/Time	4
Time in Real Time Mode	4
Run Information	
Run Initiation Time	5
Run Termination Time	5
Account Number	3,5,9,15
Project ID	3,5,9
Estimated Resource Usage	5,9
Run Type	9
Start Time	9
Deadline Time	9
Priority	9
RUNID (Original/Generated)	9,14,15,21,24

NOTE: Words 0,25,26,27 of all entries are identical and contain the entry type, the date and time that the entry was recorded, and the identity of the run.

Table 2. Log Entries by Information Items (continued)

Group	Name	Type	Code
1	432	Drum	F4
2	1782	Drum	F17
3	8440	Disk	F40
4	8414-8424	Disk	F14
5	U16	Tape	16N
6	FASTRAND	Drum	F2
7	8C-8C9	Tape	8CB-8C9
8	-	-	-
9	-	-	-
10	-	-	-

Table 3. Device Groups 1 - 10

Group	Transfer Time/Word	Latency Time	Seek Time	Startup Time	Access Time
1	4.1	4,125	-	-	4,300
2	4.1	1,700	-	-	17,000
3	7.0	12,500	7,500	-	35,000
4	14.0	12,500	12,500	-	60,000
5	20.0	-	-	3,500	-
6	400.0	35,000	35,000	-	92,000
7	625.0	-	-	4,500	-

NOTE: All times are expressed in microseconds. The average Access Time (Latency + Seek) is given by Univac [4]. The previous times are those used at the University of Maryland for billing purposes.

Table 4. Overhead Times for I/O Devices

Control Card	Charge (Instruction Cycles)
ADD	3,300
ASG	4,200
BRKPT	13,200
CAT	3,200
CHKPT	10
CKPAR	10
COG	80
DATA	10
ELT	10
EN	10
ENDF	10
EOF	10
FILE	4,500
FIN	10
FREE	2,300
FURPUR	10
HDG	30
MODE	70
MSG	30
PASSWD	10
PMD	10
QUAL	50
RSPAR	10
RSTART	10
RUN	10
START	4,300
SYM	1,100
USE	70
XQT	10

Table 5. Control Card Charges

ER Name	Octal Function Code	Description	Cost (Instruction Cycles)
ABORT$	12	Abort run	1500
ABSAD$	30	Access to downed main storage	2000
ACLST$	141	Define ASCII Control statements	2300
ACSF$	140	Dynamic Control statement request ASCII	5300
ACT$	147	Activity activation	600
ADACT$	154	CADD$ and ACT$ (ESI only)	Not Implemented
ADED$	161	Dedicate this activity to a specific processor	600
APCHCA$	77	ASCII punch control alternate	5000
APCHCN$	75	ASCII punch control	5000
APNCHA$	73	ASCII punch alternate	2400
APRINT$	70	ASCII print	2400
APRNTA$	71	ASCII print alternate	2400
APRTCA$	76	ASCII print control alternate	6400
APRTCN$	74	ASCII print control	6100
APUNCH$	72	ASCII punch	2400
AREAD$	166	ASCII read	3200
AREADA$	167	ASCII read alternate	2400
ATREAD$	170	ASCII print and read	4200
AWAIT$	134	Wait for other activities to terminate	1000
BANK$	160	Acquire Bank Descriptor Index	2400
BBEOF$	36	Set block buffering end-of-file	2300
CADD$	57	Add communications buffer	1000
CEND$	100	Terminate contingency status	35
CGET$	56	Get communications buffer	1000
CJOIN$	151	Expand communications buffer pool	2000
CLIST$	153	User access to control statements	2300
CMD$	51	Dial communications line	2000
CMH$	52	Hang-up communications line	2000
CMI$	47	Initiate communications input	1000
CMO$	50	Initiate communications output	1000
CMS$	45	Line terminal initiation	2000
CMSA$	53	Initiate communications input and output	1300
CMT$	46	Terminate communications line	2000
COM$	10	Console output and solicited input	1000
COND$	66	Retrieve condition word	90
CPOOL$	55	Create communications buffer pool	2300
CREL$	152	Release communications buffer pool	2000
CRTN$	35	Remove activity from contingency and return	90
CSF$	17	Control statement function	5300
C$TS	123	Clear test and set and notify EXEC	500
C$TSA	124	Clear test and set and activate	500
C$TSQ	122	Clear test and set and queue	1500
DACT$	150	Activity deactivation	1000
DATE$	22	Retrieve time and date	90
EABT$	26	Error terminate run	1500
ERR$	40	Error terminate activaty	1200

Table 6. Executive Request Charges

ER Name	Octal Function Code	Description	Cost (Instruction Cycles)
EXIT$	11	Normal activity termination	1000
EXLNK$	173	Return to calling reentrant processor or main program	450
FACIL$	114	Retrieve file assignment information	2300
FACIT$	143	Retrieve file assignment information	2300
FITEM$	32	Retrieve file assignment information	2300
FORK$	13	Create new activity	1250
IALL$	101	Register contingency routine	2000
II$	27	Wait for unsolicited console input	1000
INT$	33	Asynchronously interrupt named activity	200
IO$	1	Initiate I/O	1800
IOARB$	21	Initiate arbitrary device I/O	4100
IOAXI$	20	Exit and initiate arbitrary device I/O with interrupt activity	2700
IOI$	2	Initiate I/O with interrupt activity	4100
IOW$	3	Initiate I/O and wait	2700
IOWI$	24	Initiate I/O with interrupt activity and wait	5450
IOXI$	25	Exit and initiate I/O with interrupt activity	2700
LABEL$	31	Manipulate tape labels	2700
LCORE$	44	Release program storage	2700
LINK$	171	Attach to reentrant processor	375
LOAD$	111	Load program segment	12100
MCORE$	43	Acquire program storage	11200
MCT$	41	Retrieve master configuration table	9600
MSCON$	125	Master file directory manipulation	2250
NAME$	146	Name an activity	2400
NRT$	62	Terminate real-time status	100
OPT$	63	Retrieve options	90
PCHCA$	165	Punch control alternate	5000
PCHCN$	164	Punch control	5000
PCT$	64	Program control table retrieval	600
PFD$	106	Delete an element from a program file	2300
PFI$	104	Insert an element into a program file	2300
PFS$	105	Find an elemdnt in a program file	2300
PFUWL$	107	Update next program file write location	2300
PFWL$	110	Obtain next program file write location	2300
PNCHA$	145	Punch alternate	2400
PRINT$	16	Print	2400
PRINTA$	144	Print alternate	2400
PRTCA$	155	Print control alternate	5600
PRTCN$	137	Print control	5600
PSR$	157	Processor state register control	90
PUNCH$	130	Punch	2400
READ$	15	Read	2400
READA$	42	Read alternate	2400
RLINK$	172	Attach to a reentrant processor	350
RLIST$	175	Reentrant processor registration	5000
ROUTE$	133	Line terminal transfer	2300
RT$	61	Establish real-time status	100
SETBP$	156	Set 1110 programmable breakpoint register	2000
SETC$	65	Set condition word	90

Table 6. Executive Request Charges (continued)

ER Name	Octal Function Code	Description	Cost (Instruction Cycles)
SNAP$	126	Snapshot dump	10800
TDATE$	54	Retrieve time and date	90
TFORK$	14	Create new activity with timed wait	1500
TIME$	23	Retrieve time of day	90
TINTL$	136	Initialize tape file to beginning of first reel	4900
TREAD$	102	Print and read	4200
TSQCL$	113	De-register test and set queueing for programs	90
TSQRG$	121	Register test and test queueing for program	90
TSWAP$	135	Swap reels of tape file	5000
TWAIT$	60	Timed wait	1500
UNLCK$	67	Terminate interrupt activity status	90
UNLNK$	174	Return to main program from reentrant processor	350
WAIT$	6	Wait for completion of I/O request	1350
WANY$	7	Wait for any I/O completion	1350

Table 6. Executive Request Charges (continued)

Bibliography

[1] Agrawala, A. K., Bryant, R. M., and Mohr, J. M., "An Approach to the Workload Characterization Problem", University of Maryland, Department of Computer Science, Technical Report TR-410, September 1975.

[2] UNIVAC 1100 Series Operating System Programmer Reference, UP-4144, Sperry Rand Corporation, 1973, 1975.

[3] Agrawala, A. K., and Frederickson, G. N., "Scheduling in EXEC-8 Operating Systems", University of Maryland, Department of Computer Science , Technical Report TR-377, May 1975.

[4] UNIVAC 1110 System Description, UP-7841, Sperry Rand Corporation, 1971.

[5] Software Release Documentation of the UNIVAC 1100 Series Executive, Release 32.R1., Sperry Rand Corporation, June 20, 1975.

APPENDIX A

Description and Format of the Log Entries *

CONTROL STATEMENT LOG ENTRIES

Log entries specified by the @LOG control statement are placed in the master log file in the order in which they occur. The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (1)*	*nbr-of-wds-in-entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224-word-block*
	message (22-word maximum)					
25	*date-and-time-of-log-entry*					
26						
27	*run-id**					

**Run-id word is EXEC 8 if the entry (and the block) pertains to the Executive rather than a specific run.*

FACILITY USAGE LOG ENTRIES

Whenever the configuration of a run is changed by assigning or freeing a tape or arbitrary device file, an entry is made in the master log file. The entry format is:

Word	S1	S2	S3	S4	S5	S6	
0	*entry-type (2)*	*nbr-of-wds-in-entry*	*system-indicator*	unused	unused	*nbr-of-entries-in-a-224-word-block*	
1	*subsystem-nbr*		*unit-nbr*		*equipment-code**		Entry 1
2	*date-and-time-of-@ASG-or-@FREE*						
3							Entry 2
4							
24	*SUPS + voluntary-delay-time*						
25	*date-and-time-of-log-entry*						
26							
27	*run-id*						

See Appendix E for equipment codes.

CATALOGUED MASS STORAGE FILE USAGE ENTRY

Whenever a catalogued file is created, assigned, or freed (using a @FREE control statement or at run termination) a log entry is created and subsequently inserted into the master log. The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (3)*	*nbr-of-wds -in-entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224-word-block*
1	*qualifier*					
2						
3	*filename*					
4						
5	*project-id*					
6						
7	*account-number*					
8						
9	*equip-code*	*flag 2*	*current-assigns*		*absolute-F-cycle*	
10	*date/time-of-FREE-or-0*					
11	*date/time-of-cataloguing*					
12	*date/time-of-ASG,A-or-0*					
13	*Words 13 through 20 contain count of file granules at time of log entry creation which exist on mass storage devices having equipment codes 030 through 037. Equipment codes are specified in Appendix E.*					
14						
15						
16						
17						
18						
19						
20						

Word	
21 ↓ 24	
24	*SUPs + voluntary-delay-time*
25	*date-and-time-of-log-entry*
26	
27	*run-id*

where:

Word 9

current-assigns — The number of current assignments for this file when the log entry is created

flag 2

- 040 – Position granularity
- 020 – Private file
- 010 – File is being dropped
- 04 – File was or is being assigned with exclusive use
- 02 – Write only file
- 01 – Read only file

PROGRAM TERMINATION LOG ENTRY

For each program in the run, termination information is entered in the master log. The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (4)*	*nbr-of-wds -in-entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224 word-block*
1 2	*program-name*					
3 4	*version-name*					
5	*program-termination-date/time*					
6	*CPU-time (1106/1108) or-SRC-primary-storage (1110)*					
7	*–0 (1106/1108) or-SRC-extended-storage (1110)*					
8	**ER-and-control-card-charge (200 usec intervals)*					

*Words 9 through 18 contain I/O transfer counts for Group 1 through Group 10 I/O devices

Word		
19	*SUPs (200 usec intervals)*	
20	*CBSUPs (core-block-SUPs)*	
21	*voluntary-delay-time (200 usec intervals)*	
22	*time-in-real-time-mode*	
23	*unused*	*last-reentry-address*
24	*condition-word*	
25	*date-and-time-of-log-entry*	
26		
27	*run-id*	

RUN TERMINATION LOG ENTRY

At the completion of each tun, termination information is entered in the master log. The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (5)*	*nbr-of-wds -in-entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224-word-block*
1 2	*account-number*					
3 4	*project-id*					

Word			
5	*run-initiation-time (TDATE$ format)*		
6	*run-termination-time (TDATE$ format)*		
7	*cards-in*		*cards-out*
8	*priority*	*page-count-using-standard-page-size*	
9	*estimate-run-time-in-SUP's*		
10	*actual-run-time-in-SUP's*		
11	*core-block-SUP's*		
12–19	*Words 12 through 19 contain the parameters Tracks • (SUPs + voluntary-delay-time) for each of the mass storage device types. Tracks applies to temporary files and expansion of catalogued files assigned to the run.*		
20–21			
22	*total-number-of-sectors-allocated/released*		*total-number-of-allocation/release-calls*
23	*granule-table-r/w*		*total-number-of-directory-control-calls*
24	*total-number-of-PRINT$-pages*		*PRINT$-pages-since-last-BRKPT*
25	*date-and-time-of-log-entry*		
26			
27	*run-id*		

where:

Word 23

granule-table-r/w — A count of the read/write operations used to maintain granule tables.

total-directory-control-calls — The total count of references to the Executive cataloguing routines to support all catalogued files in the run.

I/O ERROR LOG ENTRY

A record of all I/O errors is kept in the master log. A count of valid references to a unit is maintained in main storage and when an error occurs, this information along with the error information is placed in the master log. The reference count is cleared to zero at that time. Note that after a predetermined number of retries (number of retries is device dependent), all of which fail, the operator is notified and operator intervention is required. The entry format is:

<table>
<tr><th>Word</th><th>S1</th><th>S2</th><th>S3</th><th>S4</th><th>S5</th><th>S6</th></tr>
<tr><td>0</td><td>entry-type (6)</td><td>nbr-of-wds-in-entry</td><td>system-indicator</td><td>unused</td><td>unused</td><td>nbr-of-log-entries-in-224-word-block</td></tr>
<tr><td>1</td><td>MSA-device-flag</td><td>subcode</td><td>equipment-code</td><td>unit-nbr</td><td colspan="2">subsystem-nbr</td></tr>
<tr><td>2</td><td colspan="6">EI-status-word (1)</td></tr>
<tr><td>3</td><td colspan="6">EI-status-word (2)-or-zero</td></tr>
<tr><td>4</td><td colspan="6">EI-status-word (3)-or-zero</td></tr>
<tr><td>5</td><td colspan="2">retry count. negative if retry failed, positive if retry passed</td><td colspan="4">nbr-of-refs-since-last-error</td></tr>
<tr><td>6</td><td colspan="3"></td><td>IOC-or-MSA-nbr</td><td>IOC-or-MSA-channel-nbr</td><td>CPU-channel-nbr</td></tr>
<tr><td>7</td><td colspan="6">EF-word (1)</td></tr>
<tr><td>8</td><td colspan="6">EF-word (2)</td></tr>
<tr><td>9 → 20</td><td colspan="6">Up to 12 additional EF words</td></tr>
<tr><td>20 → 24</td><td colspan="6"></td></tr>
<tr><td>24</td><td colspan="6">reel-nbr (tape-only) or-pack-id (disc only)</td></tr>
</table>

Word						
25	*date-and-time-of-log-entry*					
26						
27	*EXEC 8*					

where:

Word 1

MSA-device-flag	A zero indicates a non-MSA device. A non-zero indicates an MSA device.
equipment-code	See Appendix E.

CONSOLE LOG ENTRIES

Each console message is placed in the master log. At run termination, every message pertaining to the run is printed at the end of the program listing. The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (7)*	*nbr-of-wds-in-entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224-word-block*
1					msg-nbr	
2 ↓ 24	*message (23-word maximum)*					
25	*date-and-time-of-log-entry*					
26						
27	*run-id*					

CHECKPOINT LOG ENTRY

When a checkpoint is used in a run, an entry is made in the master log with pertinent information concerning the checkpointed run. The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (8)*	*nbr-of-wds -in-entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224-word-block*
↓	*message (24-word maximum)*					
25	*date-and-time-of-log-entry*					
26						
27	*run-id*					

RUN INITIATION LOG ENTRY

When a run is opened, an entry is made in the master log with the pertinent information concerning the run. The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (9)*	*nbr-of-wds-in entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224-word-block*
1	*A*	*priority*	*start-time (in minutes)*		*deadline-time (in minutes)*	
2	*estimated-pages-out*			*estimated-card-out*		
3	*run-id (new or generated)*					
4	*run-id (old or original)*					

Word				
5 6	*project-id*			
7 8	*account-number*			
9	*sequence-id*			
10	run type			*estimated-run-time-(SUP's)*
11	*device-association*			
↓				
25	*date-and-time-of-log-entry*			
26				
27	*run-id*			

Word 1

A — The possible values are:

- 010 — T option is specified on @RUN control statement
- 004 — P option is specified on @RUN control statement
- 002 — C option is specified on @RUN control statement
- 001 — S option is specified on @RUN control statement

Word 9

sequence-id — If the run is from a batch input device, this parameter is the run-id of the preceding run from the same device. If the run was scheduled via @START, this parameter is 0.

Word 10

run type — The possible types are
- 4 — demand
- 5 — deadline batch
- 6 — batch

Word 11

device-association — This parameter is the Fieldata name of the device which read the run, or, in the case of a run scheduled via @START, it is the Fieldata name of an onsite input device.

CONSOLE REPLIES LOG ENTRY

Replies to console type and read messages are placed in the master log. The replies as well as the type and read messages are printed at the end of the program listing. The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (10)*	*nbr-of-words-in-entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224-word-block*
1					*msg-nbr*	
	message (11-word maximum)					
25	*date-and-time-of-log-entry*					
26						
27	*run-id*					

LOG KEYIN ENTRY

The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (11)*	*nbr-of-wds-in-entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224-word-block*
↓	*message (9-word maximum)*					
25	*date-and-time-of-log-entry*					
26						
27	*EXEC 8*					

UNSOLICITED KEYIN LOG ENTRY

The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (12)*	*nbr-of-wds-in-entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224-word-block*
1				*actual-keyin-in-Fieldata-format (left-justified)*		
↓	*message (23-word maximum)*					
25	*date-and-time-of-log-entry*					
26						
27	*EXEC 8*					

TAPE LABELING LOG ENTRY

When the tape labeling feature of the Executive is used, log entries are made in the master log. These entries contain pertinent information concerning allocation and release of tape reels, and errors encountered during tape labeling. The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (13)*	*nbr-of-wds-in-entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224-word-block*
	message (24-word maximum)					
25	*date-and-time-of-log-entry*					
26						
27	*run-id*					

SYMBIONT END OF PROCESSING LOG ENTRY

When a symbiont finishes processing an input or output file, an entry is made in the EXEC chain on the master log with pertinent information concerning the processed file. The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (14)*	*nbr-of-wds-in-entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224 word-block*
1	*file-type*	*equipment-type*				
2	*symbiont-name*					
3	*line-count-or-cards-in/out*					
4	*run-id-of-associated-run*					

25	*date-and-time-of-log-entry*
26	
27	*EXEC 8*

where:

Word 1

file type has value of:

01 – input cards

02 – output cards

03 – output pages

SYMBIONT START OF PROCESSING LOG ENTRY

When a symbiont begins processing an input or output file, an entry is made in the EXEC chain on the master log with pertinent information regarding the file to be processed. The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (15)*	*nbr-of-wds-in-entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224-word-block*
1	*file-type*	*equipment-type*				
2	*symbiont-name*					
3						
4	*run-id-of-associated-run*					
5 6	*account-number*					

25	*date-and-time-of-log-entry*
26	
27	*run-id*

where:

Word 1

file-type has value of:

- 01 — input cards
- 02 — output cards
- 03 — output pages

PROGRAM INITIATION LOG ENTRY

When a program is initiated, a log entry is made to record pertinent values that are cumulative during the run. The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (16)*	*nbr-of-words -in-entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224-word-block*
1–2	*program-name*					
3–4	*version-name*					
5	*program-initiation-date/time*					
6	*CPU-time (1106/1108) or-primary-storage-SRC (1110)*					
7	*−0 (1106/1108) or-extended-storage-SRC (1110)*					
8	*ER + control-statement-charge-(200 usecs)*					

Word	
9 ↓ 18	I/O-transfer-count – group-1-devices ⋮ I/O-transfer-count – group-10-devices
19	SUPs – (200 usecs)
20	CBSUPs – (core-block-SUPs)
21	voluntary-delay-time (200 usecs)
22	
23	unused
24	condition-word
25	date-and-time-of log-entry
26	
27	run-id

RUN TERMINATION SUPPLEMENT LOG ENTRY

The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	entry-type (17)	nbr-of-wds-in-entry	system-indicator	unused	unused	nbr-of-log-entries-in-224-word-block
1	CPU-time-(1106/1108)-or-SRC-primary-storage-(1110)					
2	-0-(1106/1108)-or-SRC-extended-storage-(1110)					
3	ER-&-control-card-charge-(200 usecs)					
4 ↓	I/O-transfer-count-group-1-devices					
13	I/O-transfer-count-group-10-devices					
14	voluntary-delay-time (200 usecs)					
15 ↓	CAU-time (1110 only)					
25	date-and-time-of-log-entry					
26						
27	run-id					

RECOVERY CLOSE-OUT LOG ENTRY

When a system recovery is performed before a run has gone through run termination accounting, the sequence of log entries for that run may not contain the type 17 log entry which effectively closes out the set of log entries for that run. In this case, the following log entry is added to the set of log entries at the time of system recovery. This log entry effectively closes an open set of log entries. The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (18)*	*nbr-of-wds -in-entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224-word-block*
1 → 6	"SYSTEM HAS ABORTED. LOG RECOVERED AT: hhmmss"					
7						
8 → 25	*date-and-time-of-log-entry*					
26						
27	*run-id*					

COOPERATIVE ACCOUNTING

When a PRINTS or PUNCHS file is BRKPTed or closed, or when an alternate print or punch file is closed, a log entry will be inserted into the log. The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (19)*	*nbr-of-wds-in-entry*	*system-indicator*			*nbr-of-log-entries-in-224-word-block*
1	*PRLEN*		*PRCUR*		*PRTOP*	*PRBOT*
2	*Print/Punch-control-calls*			*Pages/Cards*		
3	*File-Type-indicator*			*Pages/Cards since last BRKPT*		

4 5	*FILE NAME*
↓	
25	*date-and-time-of-log-entry*
27	*run-id*

where:

Word 1

PRLEN	–	Current length of pages less bottom margin
PRCUR	–	Current position on page
PRTOP	–	Current top margin
PRBOT	–	Current bottom margin

Word 3

File Type indicator	–	2 = punch file, 3 = print file

SYMBIONT CLOSE-OUT LOG ENTRY

When the last output file for a run has been processed, an entry is made in the EXEC log chain. The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (21)*	*nbr-of-wds -in-entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224-word-block*
1	*run-id-of-associated-run*					
↓						
25	*date-and-time-of-log-entry*					
26						
27	*EXEC 8*					

EXEC SEGMENT VALIDATION LOG ENTRY

When an error is encountered while loading an EXEC segment a log entry is made and subsequently placed in the Master Log. The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (22)*	*nbr-of-wds -in-entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224-word-block*
1	*mass-storage-address-of-segment*					
2	*keyword-or-checksum-received*					
3	*'$FUNC$'-or-checksum-expected*					
4	*ANS = (A,B,or G)*			*number-of-attempts-to-load-segment*		
5	*segment-name*					
6						
↓						
25	*date-and-time-of-log-entry*					
26						
27	*EXEC 8*					

A type 22 log entry is also generated when an error is encountered reading a mass storage master bit table. In this case the entry format is as follows:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (22)*	*nbr-of-wds-in-entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224 word-block*
1	*mass-storage-address-of-MBT*					
2	*checksum-received*					
3	*checksum-expected*					
4	*pack-id-or-0*					
5	*'MBTERR'*					
25	*date-and-time-of-log-entry*					
26						
27	*EXEC 8*					

SOFTWARE ERROR LOG ENTRY
FREL Ø57 STOP

Word						
0	Entry-type (24)	NBR-of-wrds. in-entry	System Indicator		NBR-of-log entries	
1		Subcode (0001)				
2	'FREL'					
3	Qualifier					
4						
5	Filename					
6						
7	User run ID					
10	Assign options taken from file item					
11	Initial Reserve	Maximum granules				
12	Largest trick referenced	Highest granule assigned				
13	Subsystem and unit number	Rel cycle number	Absolute Cycle number			
14	Pack ID from FATEL (if DISC)					
15	CALLER					
16		MET flag	FRSEC call	Address of unit table in FATEL		
17	Equip. type	0=fast 1=drum			non-zero variable	00=TRK 40=POS
20		Count of Granules Released				
21	IDL	Location				
22	Number of trks. released		Number of position now available			
23	Track and position to release					
24	Granule address being released					
25	Original request limits					
26	Bit pattern for position where					
27	Release cannot be made					
30	Request limits at time of malfunction					
31	Date and time of log entry					
32						
33	'EXEC 8'					

SOFTWARE ERROR LOG ENTRY
FREL Ø56 STOP

Word					
0	Entry-type (24)	NBR-of-wrds. in-entry	System Indicator		NBR-of-log entries
1					Subcode (0002)
2	'Fall'				
3	Qualifier				
4					
5	Filename				
6					
7	User run ID				
10	Assign options taken from file item				
11	Initial Reserve			Maximum granularity	
12	Largest track referenced			Highest granule assigned	
13	Subsystem/unit number			Rel cycle number	Absolute Cycle number
14	Pack ID from FATEL				
15	CALLER				
16		MBI flag	FASEC call	Address of unit table in FATEL	
17	Equip. type	0=fast 1=drum	0=trk. 1=pos.	WAD flag	# of contigous trks.
20	Track taken from MBT			POS taken from MBT	
21	Most availability			Least availability	
22	Track taken from FATBL			POS taken from FATBL	
23	Next position			Next track	
24	If split request it's value				
25	Pack-ID Index			0=fixed 1=remove	
26	Previously allocated granule				
27	Trks. available from unit table at malfunction			Pos available from unit table at malfunction	
30	Word-2-of request packet				
31	Date and time				
32					
33	'EXEC 8'				

Software-detected Error Log Entry

File conflict registering removable disc

Word						
Ø	24	#WDS	SYS IND	Ø	Ø	# OF LE
1						ØØØ3
2	'UPROAR' or 'UPREG'					
3	PACKID					
4	QUALIFIER					
5						
6	FILE NAME					
7						
8	PROJECT ON THE PACK					
9						
10	ACCOUNT ON THE PACK					
11						
12	TIME OF LAST REFERENCE					
13	TIME OF CATALOGING					
14	F-CYCLE		Ø	# TIMES ASSG'D		
15	EQUIP		SUBCODE			
16	PROJECT IN THE MFD					
17						
18	ACCOUNT IN THE MFD OR Ø					
19						
20	TIME OF LAST REFERENCE OR Ø					
21	TIME OF CATALOGING OR Ø					
22	EQUIP		SUBCODE	# TIMES ASSG'D OR Ø		
23				ERROR ADDRESS		
24						
25	DATE AND TIME OF LOG ENTRY					
26						
27	EXEC 8					

Software-detected Error Log Entry

Removable disc directory error

Ø	24	# WDS	SYS IND	Ø	Ø	# of LE
1					ØØØ4	
2	'UPROAR'					
3	PACKID					
4	INITIAL DIR. TRACK ADDRESS					
5	STATUS	FUNC				
6	ACCESS WORD					
7	DISC/DRUM ADDRESS					
8				ERROR ADDRESS		
9	FIRST WORD INITIAL DAS					
1Ø	LAST WORD INITIAL DAS					
11	TRK ADDR WITHIN DAS					
25	DATE & TIME OF LOG ENTRY					
26						
27	EXEC 8					

Software-detected Error Log Entry

Removable disc pack cannot be registered

Ø	24	# WDS	SYS IND	Ø	Ø	#LE
1					ØØØ5	
2	'UPROAR'					
3	PACKID					
4	INITIAL DIR. TRACK ADDRESS					
5	STATUS	FUNC				
6	ACCESS WORD					
7	DISC/DRUM ADDRESS					
8				ERROR ADDRESS		
9	FIFPTS					
10	FIPTLK			FATULK		
25	DATE & TIME OF LOG ENTRY					
26						
27	EXEC 8					

Software-Detected Error Log Entry

C/SP illegal function or communications error

Ø	24	# WDS	SYS IND	Ø	Ø	# OF LE
1	SUBSYTEEM		LGTYPE	C/SP #	ØØØ73	
2	QCCIØ					
3	QCCI1					
4	QCIO DATA					
5	QCIO DATA...					
25	DATE & TIME OF LOG ENTRY					
26						
27	EXEC 8					

LGTYPE - Ø - C/SP initiated log entry
         1 - Illegal function from the C/SP
         2 - Communications Error

CHECKPOINT SUP CHARGE LOG ENTRY

Whenever a CKPT operation has been completed, a log entry is created and subsequently placed in the Master Log
The entry format is:

Word	S1	S2	S3	S4	S5	S6
0	*entry-type (25)*	*nbr-of-wds -in-entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224 word-block*
1	*origin-of-CKPT:0=ER CSF$, 1=control-card, 2=CK-keyin*					
2–11	*Words 2 through 11 contain I/O transfer count accumulations during the CKPT operation of devices in Groups 1 through 10.*					
12	*ER-and-control-card-charge-accumulation-during-CKPT*					
13	*voluntary-delay-time-accumulation-during-CKPT*					
14	**CPU-time (1106/1108) or-SRC-primary-storage (1110)*					

15	*—0 (1106/1108) or-SRC-extended-storage (1110)
16	*SUP-accumulation-during-CKPT*
17	*CBSUP-accumulation-duringCKPT*
↓	
25	*date-and-time-of-log-entry*
26	
27	*run-id*

**These parameters specify CPU time or SRC's which have accumulated during the CKPT operation.*

RESTART SUP CHARGE LOG ENTRY

/henever a RSTRT operation has completed, a log entry is made and subsequently inserted into the Master Log. he entry format is:

Vord	S1	S2	S3	S4	S5	S6
0	*entry-type (26)*	*nbr-of-wds -in-entry*	*system-indicator*	unused	unused	*nbr-of-log-entries-in-224-word-block*
1–10	*Words 1 through 10 contain I/O transfer count accumulations during the RSTRT operation of devices in Groups 1 through 10.*					

11	*ER-and-control-card-charge-accumulation-during-RSTRT*
12	*voluntary-delay-time-accumulation-during-RSTRT*
13	**CPU-time (1106/1108) or-SRC-primary-storage (1110)*
14	**–0 (1106/1108) or-SRC-extended-storage (1110)*
15	*SUP-accumulation-during-RSTRT*
16	*CBSUP-accumulation-during-RSTRT*
25	*date-and-time-of-log-entry*
26	
27	*run-id*

**These parameters specify CPU time or SRCs which have accumulated during the RSTRT operation.*

Word	S1	S2	S3	S4	S5	S6	
Ø	ENTRY-TYPE (27)	NBR-OF-WORDS-IN-ENTRY	SYSTEM INDICATOR	UNUSED	UNUSED	NBR-OF-LOG ENTRIES-IN-224-WD-BLK	
1	STATUS WORD (IF ANY)						
2	RETRY-ONT NEG-IF-RETRY-FAIL	CAU NBR	IOAU NBR	BUFFER-OVERLAY-FLAG	INTERRUPT CODE*		
3	PSRM						Applicable to CAU/GRS or CAU/Storage Parity Only
4	PSRME						
5	PSRU						
6	PSRUE						
7	CONTENTS OF P (APPLICABLE TO CAU/GRS PARITY ONLY)						
8	CONTENTS OF P+1 (APPLICABLE TO CAU/GRS PARITY ONLY)						
9	CONTENTS OF P+2 (APPLICABLE TO CAU/GRS PARITY ONLY)						
10	CAPTURED P (APPLICABLE TO CAU/IOAU INTERFACE PARITY ONLY)						
25	DATE-AND-TIME-OF-LOG-ENTRY						
26							
27	RUN-ID						

* 0 - CAU/Storage Parity Check
1 - CAU/IOAU Interface Parity Check
2 - CAU/GRS Control Register Parity Check

Word	S1	S2	S3	S4	S5	S6
Ø	ENTRY-TYPE (27)	NBR-OF-WORDS-IN-ENTRY	SYSTEM INDICA-TOR	UNUSED	UNUSED	NBR-OF-LOG ENTRIES-IN-224-WD-BLK
1	UNUSED					
2	RETRY-ONT NEG-IF-RETRY-FAIL	CAU NBR	IOAU NBR	CHANNEL NBR	BUFFER OVERLAY FLAG	INTERRUPT CODE*
3	ORIGINAL STATUS WORD					
4	STATUS WORD AFTER FIRST RETRY					
5	STATUS WORD AFTER SECOND RETRY					
6	STATUS WORD AFTER THIRD RETRY					
7						
8						
9						
10						
25	DATE-AND-TIME-OF-IOC-ENTRY					
26						
27	RUN-ID					

* 4 - IOAU/ACR Parity
5 - IOAU/Storage Parity
6 - IOAU Channel Interface Parity

APPENDIX B

The Format of the Program Control Table (PCT) *

PROGRAM CONTROL TABLE

Program Control Table (PCT)

Word	Field	Contents
000	AA	original run identity
001	AB	generated run activity
002	AC	total accumulated run time for all completed tasks
003	AD	estimated run time (200 microseconds)
004	AE	account priority
004	AY	ABORTs indicator
004	AZ	T option
004	AX	core queue entry address
005	AL	qualifier table start (relative)
005	AH	activity name table
006	EA	program contingency mask
006	EC	program contingency packet
007	AG	not used
007	DY	log count
007	AF	ESI activity count
010	AM	total I/O request count
011	AN	data transfer count
012	AO	reserved for site use

Field	Word	Contents	
AP	013	standard unit of processing accumulation (SUP)	
AQ	014	standard unit of processing block accumulation	
	015	AR WAITs count	WAITs chain
AS	016	run start time and date (TDATE$ format)	
BB	017	XQT options (bit 25=A ----- bit 0=Z)	
BC	020	condition word	
BD	021	most recent qualifier (12 characters)	
	022	initial value will be project identity	
BI	023	account number (12 characters)	
	024		
CA	025	core quantum limit	
CC	026	total accumulated CPU time (200 microsecond increments)	

027	CI real-time activity count			CE swap lock counter	
030	IL TS segment load	CG program type	WH current program priority	WF original program priority	IK program size
CL 031	activities released via AWAITs (bit 35=ACTID35 - bit 1=ACTID1 - bit 0 always = 0)				
CM 032	mask control word of existing activities				
CN 033	activity mask of activity 1				
CO 034	activity mask of activity 2				
CP 035	activity mask of activity 3				
. . .	. . .				
DS 72	activity mask of activity 32				
DT 73	activity mask of activity 33				
DU 74	activity mask of activity 34				

DV 075	activity mask of activity 35			
076	DW relative address of last ASA		DX current activity count	
077	IE test & set chain	IP condition of run	CF maximum real-time level	IF linkage to CPOOLS reference
0100	TSASA activity status area test & set	EM core clear flag	JP granule table page flag	EN PCT abort mask
IA 0101	MC MD chain for EXEC workers (ASA) attached to user's PCT			
0102	IH check point count	IN display msg busy no term flag	IC error flag	ID deactivate count
0103	EO link to previous PCT		link to next PCT	
EQ 0104 0105	12 character USERID (Quota System)			
0106	EW test & set activity chain	not used	EY PCT relative BDT address	
ET 0107	core quantum time			
EU 0110	filename of programs PF used by			

EV 111	LOADs and PMD				
EX 112	file header table relative address				
113	IJ test and set PCT abort	IQ quarter word mode flag	not used	CD number of segments	
14	JI ESI test and set contingency	JJ ESI contingency proc number	JK ESI contingency counter	JL ESI contingency routine address	
15	WG test & set timing	MW MCR/LCR flag	WE time quantum insured flag	JA count for outstanding I/O request	
JD 16	JB PCT item chain start		JC PCT item chain end		
17	JG test & set PCT tight mode	do not used	JF PCT tight mode	JE ER deactivate chain	
0	JM ESI activity test & set	JZ RCR act account	JX number of RLISTED banks	JN ESI activity count	
1	JO RLIST test & set	JR not used	WB forced page	JY RLIST buffer link	
2	JS suspend flag		JQ buffer address for PMD		
3	JT PCT link test & set	JU initial PCT size	JW PMD flag	JH SNAPS flag	JV attached activity count
A r	program start (time and date)				

KB *0125* KC *0126*	program name (12 Fieldata characters)	
KD *0127* KE *0130*	program version name (12 Fieldata characters)	
0131	KF positions assigned type 030	KG tracks assigned type 030
0132	KH positions assigned type 031	KJ tracks assigned type 031
0133	KL positions assigned type 032	KM tracks assigned type 032
0134	KN positions assigned type 033	KO tracks assigned type 033
0135	KP positions assigned type 034	KQ tracks assigned type 034
0136	KR positions assigned type 035	KS tracks assigned type 035
0137	KU positions assigned type 036	KV tracks assigned type 036
0140	KW positions assigned type 037	KX tracks assigned type 037

0141	KY	time of last track second calculation		
0142	KZ	track seconds type 030 (scratch)		
143	LB	track seconds type 031		
144	LC	track seconds type 032		
145	LI	track seconds type 033		
146	LJ	track seconds type 034		
147	LK	track seconds type 035		
150	LF	track seconds type 036		
151	LQ	track seconds type 037		
52	LU	number sector allocation/release requests	LV	number of all allocation/releases
53	LW	number of granule table I/Os	LY	total of DRC references
54	LZ	not used		

Word				
0155	MB test & set AWAITs	BJ initial load error	HY correct printer change	MZ IIs address
0156	MA test & set mass storage accounting	ME TSQRGs flag	MF abort message print	MG CLIST buffer address (relative)
0157	TSSLC test & set swaplock	MH swaplock stop	MK ABORTs flag	MI ABORTs address
MJ 0160	CGYTS test & set	CGYSTA stay in cont mode on ER flag	CGYMOD PGM cont mode flag	MG contingency queue pointer
NA 0161	user BDP			
0162	NB INFOR CLISTs buffer address		NC re-entry point for ABORTs	
ND 0163	user breakpoint word 1			
0164	ACQTS test & set ACQLST	not used	ACQLST asynchronous contingency queue	
0165	NJ not used	NL allow ERs in cont mode	NI program type	NF control bank BDI
0166	not used			
0167	not used			
0170	CB core quantum size		NK activity long wait count	

NM 0171	user breakpoint word 2			
0172	NQ test & set breakpoint	PE core priority	NN enter OLM errors	NP relative address of common bank list buffer
0173	NV dynamic bank load chain TS DBLC	NS realtime position flag	PK CHKPT/RSTRT busy lock	NW dynamic bank load chain (DBLC) pointer
SUPS 0174 205	I/O transfer standard unit of processing charges ten words indexed by I/O group number (1-10)			
NX 206	ER standard unit of processing charges			
NY 207	voluntary delay time			
NZ 210	standard unit of processing quantum exceed wall time			
NT 211	length of time a program is real-time			
PF 12	TSNAM1 test & set	RSB run using RSIs	not used	swaplock trace area address
13	KDACT test & set			RLQ 31 word buffer for Quota system
14	SLRUNT test & set CPU time			NE packet address for PGM/ACT CRTNs

	0215	TSACS Test & Set			RSTIP restart-in-progress flag
GRACET	0216	grace time for max time contingency			
SIPTIM	0217	swap initiation time per task (SYSBAL)			
	0220	CQUEQ test & set			CQUEQ queue of contingency queued by CQUEUE$
	0221	test & set program TRMRG$ queue	TRMGMD program TRMRG$ mode flag		program TRMRG$ queue
	0222	not used			
NG	0223–0224	primary storage reference count (2 words)			
NH	0225–0226	extended storage reference count (2 words)			
PTIMEN	0227	base time storage monitor			
PRTIMM	0230	time reload, storage monitor anti-thrash			
PSIZEM	0231	program size (primary)			program size (extended)

PB 0232	SUPs at initial load or reload			
PC 0233	primary SRC at initial load or reload			
PD 0234	extended SRC at initial load or reload			
JSTK 0235	1100 Jump Stack save buffer pointer			
236			EMBUF PCT buffer for ERRFO lockup	
237	not used			
240	not used			
NAM 241	test & set facility	FI lock	number of names	link to next name list
42	filename (12 Fieldata characters)			
43				
44	link to external name			link to file description buffer

. . . .	*Initial facility item - 62 words. First and last words (PCTNAM, PCTBCK) are used for control. Middle 60 words contain up to 20 3-word ASG and USE name items. All USE entries come first, followed by ASGs. If more than 20 items, this fixed buffer is chained to 63-word buffers, which hold the rest of the descriptor entries. The format of these descriptors is shown here as the first entry in the list.*			
PCTBCK *0336*	MV hold queue remov attempt counter	PCTMX tape assign maximum	PCFTAS number of current tape assigns	relative link to precedent = 0 for list
TIPS1 *0337*				
0340	not used			
0341	not used			
BUFB *0342*	buffer for PCT chaining (5 words)			
. . . .	. . . .			
BUFA *0347*				
0350				

351	4		link to first 4 word buffer or 0
352	8		link to first 8 word buffer or 0
353	16		link to first 16 word buffer or 0
54	32		link to first 32 word buffer or 0
55	64		link to first 64 word buffer or 0
56	128		link to first 128 word buffer or 0
57	256		link to first 256 word buffer or 0
60	512		link to first 512 word buffer or 0

NBS-114A (REV. 11-77)

U.S. DEPT. OF COMM. BIBLIOGRAPHIC DATA SHEET	1. PUBLICATION OR REPORT NO. NBS SP 500-43	2. Gov't Accession No.	3. Recipient's Accession No.

4. TITLE AND SUBTITLE
COMPUTER SCIENCE & TECHNOLOGY:
A GUIDE TO MAJOR JOB ACCOUNTING SYSTEMS:
The Logger System of the UNIVAC
1100 Series Operating System

5. Publication Date December 1978

6. Performing Organization Code Univeristy of Marylan

7. AUTHOR(S) J. M. Mohr, A. K. Agrawala, J. F. Flannagan

8. Performing Organ. Report No. TR-434

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Department of Computer Science
University of Maryland
College Park, MD 20742

10. Project/Task/Work Unit No.

11. Contract/Grant No.
NBS 5-9017
NSF GK-41062

12. Sponsoring Organization Name and Complete Address *(Street, City, State, ZIP)*
Institute for Computer Sciences and Technology
National Bureau of Standards
Washington, DC 20234

13. Type of Report & Period Covered

14. Sponsoring Agency Code

15. SUPPLEMENTARY NOTES
Library of Congress Catalog Card Number: 78-600135

16. ABSTRACT *(A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here.)*

This report has been prepared to serve as guides to the use of Logger, the job accounting system supplied by Univac for its 1100 Series Operating System, Level 32. Logger provides a capability for the automatic collection of information that may be used both for billing a computer installations's customers on the basis of resources utilized by their programs, and for gaining useful insights into the performance characteristics of the system itself. This report describes the structure of the accounting log system, provides a description of the information contained in the log tapes, and describes how the information is gathered by the Operating System.

17. KEY WORDS *(six to twelve entries; alphabetical order; capitalize only the first letter of the first key word unless a proper name; separated by semicolons)* Computer performance analysis; computer performance measurement EXEC-8; job accounting systems; Logger system; resource utilization measurement; standard unit of processing; SUP; UNIVAC 1100 Series Operating System.

18. AVAILABILITY ☐ Unlimited

☐ For Official Distribution. Do Not Release to NTIS

☒ Order From Sup. of Doc., U.S. Government Printing Office Washington, D.C. 20402, SD Stock No. SN003-003

☐ Order From National Technical Information Service (NTIS) Springfield, Virginia 22151

19. SECURITY CLASS (THIS REPORT) UNCLASSIFIED

20. SECURITY CLASS (THIS PAGE) UNCLASSIFIED

21. NO. OF PAGES 79

22. Price $2.50

ANNOUNCEMENT OF NEW PUBLICATIONS ON COMPUTER SCIENCE & TECHNOLOGY

Superintendent of Documents,
Government Printing Office,
Washington, D. C. 20402

Dear Sir:

Please add my name to the announcement list of new publications to be issued in the series: National Bureau of Standards Special Publication 500-.

Name ______________________________

Company ______________________________

Address ______________________________

City ______________ State ______________ Zip Code ______________

(Notification key N-503)

☆ U. S. GOVERNMENT PRINTING OFFICE : 1978 281-067/330

NBS TECHNICAL PUBLICATIONS

PERIODICALS

)URNAL OF RESEARCH—The Journal of Research the National Bureau of Standards reports NBS research ıd development in those disciplines of the physical and gineering sciences in which the Bureau is active. These clude physics, chemistry, engineering, mathematics, and mputer sciences. Papers cover a broad range of subjects, th major emphasis on measurement methodology, and e basic technology underlying standardization. Also inıded from time to time are survey articles on topics closely lated to the Bureau's technical and scientific programs. As special service to subscribers each issue contains complete tations to all recent NBS publications in NBS and non-BS media. Issued six times a year. Annual subscription: mestic $17.00; foreign $21.25. Single copy, $3.00 domestic; .75 foreign.

ote: The Journal was formerly published in two sections: ction A "Physics and Chemistry" and Section B "Matheatical Sciences."

IMENSIONS/NBS

ıis monthly magazine is published to inform scientists, gineers, businessmen, industry, teachers, students, and nsumers of the latest advances in science and technology, ith primary emphasis on the work at NBS. The magazine ghlights and reviews such issues as energy research, fire otection, building technology, metric conversion, pollution atement, health and safety, and consumer product perrmance. In addition, it reports the results of Bureau proams in measurement standards and techniques, properties matter and materials, engineering standards and services, strumentation, and automatic data processing.

Annual subscription: Domestic, $11.00; Foreign $13.75

NONPERIODICALS

onographs—Major contributions to the technical literure on various subjects related to the Bureau's scientific ıd technical activities.

andbooks—Recommended codes of engineering and industal practice (including safety codes) developed in cooperaon with interested industries, professional organizations, ıd regulatory bodies.

ecial Publications—Include proceedings of conferences onsored by NBS, NBS annual reports, and other special ıblications appropriate to this grouping such as wall charts, cket cards, and bibliographies.

pplied Mathematics Series—Mathematical tables, manıls, and studies of special interest to physicists, engineers, emists, biologists, mathematicians, computer programmers, ıd others engaged in scientific and technical work.

ational Standard Reference Data Series—Provides quantitive data on the physical and chemical properties of aterials, compiled from the world's literature and critically aluated. Developed under a world-wide program codinated by NBS. Program under authority of National andard Data Act (Public Law 90-396).

NOTE: At present the principal publication outlet for these data is the Journal of Physical and Chemical Reference Data (JPCRD) published quarterly for NBS by the American Chemical Society (ACS) and the American Institute of Physics (AIP). Subscriptions, reprints, and supplements available from ACS, 1155 Sixteenth St. N.W., Wash., D.C. 20056.

Building Science Series—Disseminates technical information developed at the Bureau on building materials, components, systems, and whole structures. The series presents research results, test methods, and performance criteria related to the structural and environmental functions and the durability and safety characteristics of building elements and systems.

Technical Notes—Studies or reports which are complete in themselves but restrictive in their treatment of a subject. Analogous to monographs but not so comprehensive in scope or definitive in treatment of the subject area. Often serve as a vehicle for final reports of work performed at NBS under the sponsorship of other government agencies.

Voluntary Product Standards—Developed under procedures published by the Department of Commerce in Part 10, Title 15, of the Code of Federal Regulations. The purpose of the standards is to establish nationally recognized requirements for products, and to provide all concerned interests with a basis for common understanding of the characteristics of the products. NBS administers this program as a supplement to the activities of the private sector standardizing organizations.

Consumer Information Series—Practical information, based on NBS research and experience, covering areas of interest to the consumer. Easily understandable language and illustrations provide useful background knowledge for shopping in today's technological marketplace.

Order **above** *NBS publications from: Superintendent of Documents, Government Printing Office, Washington, D.C. 20402.*

Order **following** *NBS publications—NBSIR's and FIPS from the National Technical Information Services, Springfield, Va. 22161.*

Federal Information Processing Standards Publications (FIPS PUB)—Publications in this series collectively constitute the Federal Information Processing Standards Register. Register serves as the official source of information in the Federal Government regarding standards issued by NBS pursuant to the Federal Property and Administrative Services Act of 1949 as amended, Public Law 89-306 (79 Stat. 1127), and as implemented by Executive Order 11717 (38 FR 12315, dated May 11, 1973) and Part 6 of Title 15 CFR (Code of Federal Regulations).

NBS Interagency Reports (NBSIR)—A special series of interim or final reports on work performed by NBS for outside sponsors (both government and non-government). In general, initial distribution is handled by the sponsor; public distribution is by the National Technical Information Services (Springfield, Va. 22161) in paper copy or microfiche form.

BIBLIOGRAPHIC SUBSCRIPTION SERVICES

he following current-awareness and literature-survey bibligraphies are issued periodically by the Bureau:

ryogenic Data Center Current Awareness Service. A literature survey issued biweekly. Annual subscription: Domestic, $25.00; Foreign, $30.00.

iquified Natural Gas. A literature survey issued quarterly. nnual subscription: $20.00.

Superconducting Devices and Materials. A literature survey issued quarterly. Annual subscription: $30.00. Send subscription orders and remittances for the preceding bibliographic services to National Bureau of Standards, Cryogenic Data Center (275.02) Boulder, Colorado 80302.

U.S. DEPARTMENT OF COMMERCE
National Bureau of Standards
Washington, O.C. 20234

OFFICIAL BUSINESS

Penalty for Private Use, $300

POSTAGE ANO FEES PAIO
U.S. DEPARTMENT OF COMMERCE
COM-215

SPECIAL FOURTH-CLASS RATE
BOOK